HEAD MOTION ESTIMATION FROM FOUR FEATURE POINTS

BACKGROUND OF THE INVENTION

FIELD OF THE INVENTION

The present invention relates to systems and methods for computing head motion estimation from the facial image positions, e.g., eye and mouth corners, and, particularly, to a linear method for performing head motion estimation using four (4) facial feature points. As a special case, an algorithm for head pose estimation from four feature points is additionally described.

DISCUSSION OF THE PRIOR ART

Head pose recognition is an important research area in human computer interaction and many approaches of head pose recognition have been proposed. Most of these approaches model a face with certain facial features. For example, most existing approaches utilize six facial feature points including pupils, nostrils and lip corners are used to model a face, while others, such as reported in the reference to Z. Liu and Z. Zhang entitled "Robust Head Motion Computation by Taking Advantage of Physical Properties", *Proc. Workshop on Human Motion*, pp. 73-80, Austin, December 2000, implements five facial feature points including eye and mouth corners and the tip of the nose. In Zhang, the head motion is estimated from the five

feature points through non-linear optimization. In fact, existing algorithms for face pose estimation are non-linear.

It would be highly desirable to provide a face pose estimation algorithm that is linear, and computationally less demanding than non-linear solutions.

It would be further highly desirable to provide a face pose estimation algorithm that is linear, and relies on only four feature points such as the eye and mouth corners.

SUMMARY OF THE INVENTION

Accordingly, it is an object of the present invention to provide a head motion estimation algorithm that is a linear solution.

It is a further object of the present invention to provide a head motion estimation algorithm that is linear and utilizes four facial feature points.

It is another object of the present invention to provide a head pose estimation algorithm which relies on a head motion estimation algorithm.

In accordance with the principles of the invention, there is provided a linear method for performing head motion estimation from facial feature data, the method comprising the steps of: obtaining first facial image and detecting a head in the first image; detecting position of four points P of said first facial image where $P = \{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4\}$, and $\mathbf{p}_k = (x_k, y_k)$; obtaining a second facial image and detecting a head in the second image; detecting position of four points P' of the second facial image where

$P' = \{\mathbf{p}'_1, \mathbf{p}'_2, \mathbf{p}'_3, \mathbf{p}'_4\}$ and $\mathbf{p}'_k = (x'_k, y'_k)$); and, determining the motion of the head represented by a rotation matrix R and translation vector T using the points P and P'. The head motion estimation is governed according to an equation:

$$\mathbf{P}'_i = R\mathbf{P}_i + \mathbf{T}, \qquad \text{where } R = \begin{bmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \\ \mathbf{r}_3^T \end{bmatrix} = [r_{ij}]_{3\times 3} \text{ and } \mathbf{T} = [T_1 \quad T_2 \quad T_3]^T$$

represents camera rotation and translation respectively, the head pose estimation being a specific instance of head motion estimation.

Advantageously, the head pose estimation algorithm from four feature points may be utilized for avatar control applications, video chatting and face recognition applications.

BRIEF DESCRIPTION OF THE DRAWINGS

Details of the invention disclosed herein shall be described below, with the aid of the figure listed below, in which:

Figure 1 depicts the configuration of typical feature points for a typical head;

Figure 2 depicts the face geometry 10 providing the basis of the head pose estimation algorithm of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

In accordance with the principles of the invention, a linear method for the computation of head motion estimation from the image positions of eye and mouth

corners, is provided. More particularly, a method is provided for estimating head motion from four point matches, with head pose estimation being a special case, when a frontal view image is used as a reference position.

The method is superior to other existing methods, which require either more point matches (at least 7) or, are non-linear requiring at least 5 facial feature matches.

Generally, the method for head motion estimation is as follows: The first step is to acquire a first image I_1 and detecting the head in I_1. Then, there are detected points P corresponding to the outer corners of eyes and mouth in I_1, i.e., $P = \{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4\}$, where $\mathbf{p}_k = (x_k, y_k)$ denotes image coordinates of a point. Then, a second image I_2 is acquired with the head detected in I_2. Then, there are detected points P' corresponding the eyes and mouth and their outer corners in I_2, i.e., $P' = \{\mathbf{p}'_1, \mathbf{p}'_2, \mathbf{p}'_3, \mathbf{p}'_4\}$, where $\mathbf{p}'_k = (x'_k, y'_k)$). From P and P', the next step involves determining the motion of the head represented by a rotation matrix R and translation vector $\mathbf{T}$. It is understood that once motion parameters R and $\mathbf{T}$ are computed, the 3-D structure of all point matches may be computed. However, structure and translation may be determined only up to a scale, so if the magnitude of $\mathbf{T}$ is fixed, then the structure is uniquely determined. If the depth of one point in 3D is fixed, then T will be uniquely determined.

As mentioned, the algorithm for head pose estimation is a special case of the head motion estimation algorithm and there are two ways in which this may be

accomplished: 1) *interactive*, which requires a reference image; and, 2) *approximate*, which uses a generic (average biometric) head geometry information, also referred to as a Generic Head Model (GHM).

For the Interactive algorithm, the following steps are implemented: 1) Before using the system, a user is asked to face the camera in a predefined reference position. The reference eye and mouth corners P_0 are acquired as described in the steps above. 2) When a new image is acquired, eye and mouth corners are detected and head motion estimated as in the remaining steps indicated in the algorithm above. 3) The head rotation matrix corresponds to head pose matrix.

The Approximate algorithm requires no interaction with the user, but assumes certain biometric information is available and fixed for all the users. For example, as shown in Figure 1, there is depicted the approximate algorithm including the configuration of typical feature points for a typical head 19 in relation to a camera coordinate system 20 denoted as system C_{xyz}. In Figure 1, the points P_1 and P_3 represent the eye and mouth corners, respectively of the generic head model 19. It is understood that for the frontal view, shown in Figure 1, these points P_1 and P_3 have different depths (Z_1 and Z_3, respectively). An assumption is made that the angle τ is known, and an average value is used over all possible human heads. This is not an exact value, but pitch (tilt) angle is very difficult to compute precisely, since even the same person, when asked to look straight into camera, may tilt head differently in repeated experiments. For the fixed

angle τ, head pose may be uniquely determined from only one image of the head as will be explained in greater detail hereinbelow.

For purposes of description, it is assumed that a camera or digital image capture device has acquired two images of a model head at different positions. Let points $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$ and $\mathbf{p}_4$ denote the image coordinates of eye (points $\mathbf{p}_1$, $\mathbf{p}_2$) and mouth corners (points $\mathbf{p}_3$ and $\mathbf{p}_4$) in a first image and let $\mathbf{p}'_1, \mathbf{p}'_2, \mathbf{p}'_3, \mathbf{p}'_4$ denote the corresponding eye and mouth corner coordinates in a second image. Given these feature coordinates, the task is to determine head motion (represented by rotation and translation) between those first and second two images.

Generally, the algorithm is performed in the following steps: 1) using facial constraints, compute the three-dimensional (3-D) coordinates for the feature points from both images; and, 2) given the 3-D positions of the feature points, compute the motion parameters (rotation R and translation T matrices).

The step of computing the 3-D coordinates of feature points according to the algorithm are now described. As shown in the face geometry 10 depicted in Figure 2, features at points $\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{P}_3$, $\mathbf{P}_4$ and $\mathbf{P}'_1, \mathbf{P}'_2, \mathbf{P}'_3, \mathbf{P}'_4$ denote the 3-D coordinates of the respective eye and mouth corners in the first two images. From the face geometry, shown in Figure 2, the following properties are assumed: 1) the line segment 12 connecting points $\mathbf{P}_1\mathbf{P}_2$ is parallel to the line segment 15 connecting points $\mathbf{P}_3\mathbf{P}_4$, i.e., $\mathbf{P}_1\mathbf{P}_2 \parallel \mathbf{P}_3\mathbf{P}_4$; 2) the line segment 12 connecting points

$\mathbf{P}_1\mathbf{P}_2$ is orthogonal to a line segment connecting points $\mathbf{P}_5\mathbf{P}_6$ (where $\mathbf{P}_5$ and $\mathbf{P}_6$ are midpoints of segments $\mathbf{P}_1\mathbf{P}_2$ and $\mathbf{P}_3\mathbf{P}_4$, respectively). Numerically, these properties 1 and 2 may be written according to respective equations (1) and (2) as follows:

$$\frac{X_2 - X_1}{X_4 - X_3} = \frac{Y_2 - Y_1}{Y_4 - Y_3} = \frac{Z_2 - Z_1}{Z_4 - Z_3} \tag{1}$$

$$((\mathbf{P}_1 + \mathbf{P}_2) - (\mathbf{P}_3 + \mathbf{P}_4)) \cdot (\mathbf{P}_2 - \mathbf{P}_1) = 0 \tag{2}$$

where $\mathbf{P}_i = [X_i \quad Y_i \quad Z_i]^T$ denotes a 3D coordinates of an image point $\mathbf{p}_i$. The relation between image and the three-dimensional (3-D) coordinates of an arbitrary point P_k is given by a well-known perspective equation as follows:

$$x_k = \frac{X_k}{Z_k}, \quad y_k = \frac{Y_k}{Z_k} \tag{3}$$

Since it is well known that the structure recovery from monocular image sequences may be performed only up to a scale, one of the Z coordinates is fixed, and the other coordinates are computed in reference to this one. Hence, to simplify the computation, and without a loss of generality, it is assumed that $Z_1 = 1$. By cross-multiplying equation (1) and substituting (3) into (1), the following relations set forth in equations (4) and (5) result:

$$Z_3\left[(x_1 - x_3) - Z_2(x_2 - x_3)\right] - Z_4\left[(x_1 - x_4) - Z_2(x_2 - x_4)\right] = 0 \tag{4}$$

$$Z_3\left[(y_1 - y_3) - Z_2(y_2 - y_3)\right] - Z_4\left[(y_1 - y_4) - Z_2(y_2 - y_4)\right] = 0 \tag{5}$$

When equations (4) and (5) are set forth in matrix form, equation (6) results:

$$\begin{bmatrix} (x_1 - x_3) - Z_2(x_2 - x_3) & -(x_1 - x_4) + Z_2(x_2 - x_4) \\ (y_1 - y_3) - Z_2(y_2 - y_3) & -(y_1 - y_4) + Z_2(y_2 - y_4) \end{bmatrix} \begin{bmatrix} Z_3 \\ Z_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad (6)$$

This equation will have non-trivial solutions in Z_3 and Z_4 if and only if the determinant in equation (7) is equal to zero, i.e.,

$$\det\left(\begin{bmatrix} (x_1 - x_3) - Z_2(x_2 - x_3) & -(x_1 - x_4) + Z_2(x_2 - x_4) \\ (y_1 - y_3) - Z_2(y_2 - y_3) & -(y_1 - y_4) + Z_2(y_2 - y_4) \end{bmatrix}\right) = 0 \qquad (7)$$

Equivalently, equation (7) may be set forth as equation (8) as follows:

$$-Z_2^2 \det\left(\begin{bmatrix} (x_2 - x_3) & (x_2 - x_4) \\ (y_2 - y_3) & (y_2 - y_4) \end{bmatrix}\right) + \det\left(\begin{bmatrix} (x_1 - x_3) & -(x_1 - x_4) \\ (y_1 - y_3) & -(y_1 - y_4) \end{bmatrix}\right) - \\ Z_2\left(\det\left(\begin{bmatrix} (x_2 - x_3) & (x_1 - x_4) \\ (y_2 - y_3) & (y_1 - y_4) \end{bmatrix}\right) + \det\left(\begin{bmatrix} (x_1 - x_3) & (x_2 - x_4) \\ (y_1 - y_3) & (y_2 - y_4) \end{bmatrix}\right)\right) = 0 \qquad (8)$$

Equation (8) is a second order polynomial and it has two solutions. It is easy to verify (e.g., by substitution in (7)) that there is one trivial solution, $Z_2 = 1$, and the second solution is found as:

$$Z_2 = \frac{\det\left(\begin{bmatrix} (x_1 - x_3) & (x_1 - x_4) \\ (y_1 - y_3) & (y_1 - y_4) \end{bmatrix}\right)}{\det\left(\begin{bmatrix} (x_2 - x_3) & (x_2 - x_4) \\ (y_2 - y_3) & (y_2 - y_4) \end{bmatrix}\right)}. \qquad (9)$$

By substituting Z_2 into any of equations (4) and (5) one linear equation in Z_3 and Z_4 is obtained. Another equation

is obtained by substituting (3) into (2) and it is of the form:

$$Z_3\mathbf{p}_{h3}^T(\mathbf{P}_1-\mathbf{P}_2)+Z_4\mathbf{p}_{h4}^T(\mathbf{P}_1-\mathbf{P}_2)=\|\mathbf{P}_1\|^2-\|\mathbf{P}_2\|^2. \tag{10}$$

where $\mathbf{p}_{hi}=[x_i \quad y_i \quad 1]^T$. Z_3 and Z_4 may now be solved from equations (10) and (4)

As known, the motion of head points can be expressed according to equation (11) as:

$$\mathbf{P}_i'=R\mathbf{P}_i+\mathbf{T} \tag{11}$$

where $R=\begin{bmatrix}\mathbf{r}_1^T\\ \mathbf{r}_2^T\\ \mathbf{r}_3^T\end{bmatrix}=[r_{ij}]_{3\times 3}$ and $\mathbf{T}=[T_1 \quad T_2 \quad T_3]^T$ represent camera rotation and translation respectively. Equation (11) may now be written in terms of R and $\mathbf{T}$ as:

$$\begin{bmatrix}\mathbf{P}_i^T & \mathbf{0}^T & \mathbf{0}^T & 1 & 0 & 0\\ \mathbf{0}^T & \mathbf{P}_i^T & \mathbf{0}^T & 0 & 1 & 0\\ \mathbf{0}^T & \mathbf{0}^T & \mathbf{P}_i^T & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\mathbf{r}_1\\ \mathbf{r}_2\\ \mathbf{r}_3\\ \mathbf{T}\end{bmatrix}=\mathbf{P}_i' \tag{12}$$

From equation (12) it is observed that each point pair yields 3 equations. As the total number of unknowns is twelve (12), at least four point pairs are necessary to linearly solve for rotation and translation.

It should be understood that the elements of matrix R are not independent (i.e., $RR^T=I$), so once matrix R is solved, it may need to be corrected so that it represents the true rotation matrix. This may be performed

by decomposing R using Singular Value Decomposition (SVD) into a form $R = USV^T$, and computing a new rotation matrix according to equation (13) as follows:

$$R = UV^T. \tag{13}$$

As known, a "Head Pose" may be uniquely represented as a set of three angles (yaw, roll and pitch), or, as a rotation matrix R (given that there is a one-to-one correspondence between the rotation matrix and the pose angles). Interactive head pose estimation is equivalent to head motion estimation however, an approximate head pose estimation is described which may be simplified by decomposing it into two steps, as follows: 1) assuming that user has tilted his/her head so that both eye and mouth corners are at the same distance from the camera ($z_1 = z_2 = z_3 = z_4$), and that this is an Auxiliary Reference Position (ARP); 2) compute head pose for the ARP; and, 3) updating a pitch angle, by simply subtracting τ from its value in ARP.

The rotation matrix R may be written as follows:

$$R = \begin{bmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \\ \mathbf{r}_3^T \end{bmatrix} = \left[r_{ij} \right]_{3\times 3}$$

which satisfies the condition, $RR^T = I$, or equivalently

$$\mathbf{r}_i^T\mathbf{r}_j = \begin{cases} 0, & i \neq j \\ 1, & i = j \end{cases}. \tag{14}$$

Let $\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3, \mathbf{F}_4$ denote the 3-D coordinates of the eye and mouth corners of the reference, frontal view of the face. Then, accounting for the face geometric constraints and constraint 1) above, there is obtained the relations governed by equations 15) as follows:

$$\begin{aligned} \mathbf{F}_2 - \mathbf{F}_1 &\propto [1 \quad 0 \quad 0]^T \\ \mathbf{F}_6 - \mathbf{F}_5 &\propto [0 \quad 1 \quad 0]^T \end{aligned} \tag{15}$$

where symbol $\propto$ means "equal up to a scale" or proportional. The goal accomplished by the present invention is to find a pose matrix R that maps points $\mathbf{P}_k$ to $\mathbf{F}_k$, i.e.,

$$\begin{aligned} R(\mathbf{P}_2 - \mathbf{P}_1) &\propto [1 \quad 0 \quad 0]^T \\ R(\mathbf{P}_6 - \mathbf{P}_5) &\propto [0 \quad 1 \quad 0]^T \end{aligned}. \tag{16}$$

In terms of rows of rotation matrix, equation (16) may be written as:

$$\begin{aligned} \mathbf{r}_2^T(\mathbf{P}_2 - \mathbf{P}_1) &= 0 \\ \mathbf{r}_3^T(\mathbf{P}_2 - \mathbf{P}_1) &= 0 \\ \mathbf{r}_1^T(\mathbf{P}_6 - \mathbf{P}_5) &= 0 \\ \mathbf{r}_3^T(\mathbf{P}_6 - \mathbf{P}_5) &= 0 \end{aligned}. \tag{17}$$

From the second and fourth equation in (17), $\mathbf{r}_3$ may be computed as follows:

$$\mathbf{r}_3 = (\mathbf{P}_6 - \mathbf{P}_5) \times (\mathbf{P}_2 - \mathbf{P}_1). \tag{18}$$

The remaining components of the rotation matrix may be computed from (14) and (17) as:

$$\begin{aligned} \mathbf{r}_2 &= \mathbf{r}_3 \times (\mathbf{P}_2 - \mathbf{P}_1) \\ \mathbf{r}_1 &= \mathbf{r}_2 \times \mathbf{r}_3 \end{aligned} \tag{19}$$

From equation (19) it is straightforward to compute yaw, roll and pitch angles. The true pitch angle is then obtained by subtracting τ from its current value.

While there has been shown and described what is considered to be preferred embodiments of the invention, it will, of course, be understood that various modifications and changes in form or detail could readily be made without departing from the spirit of the invention. It is therefore intended that the invention be not limited to the exact forms described and illustrated, but should be constructed to cover all modifications that may fall within the scope of the appended claims.